EnSight parallel computing at LLNL

OVerview of the four modes of EnSight visualization. The modes are: **standalone**, **distributed serial,** d**istributed parallel (SOS),** and **distributed parallel + distributed rendering (SOS + DR).** These instructions assume a Linux or Mac OS X desktop. If you are running Windows, then please contact lc-graphics@llnl.gov for specific instructions for your use case. Windows users will have to run ceishell "by hand" to make it work, and set up tunnels also manually. To download this as a PDF click here.

Using the LLNL helper scripts for distributed modes

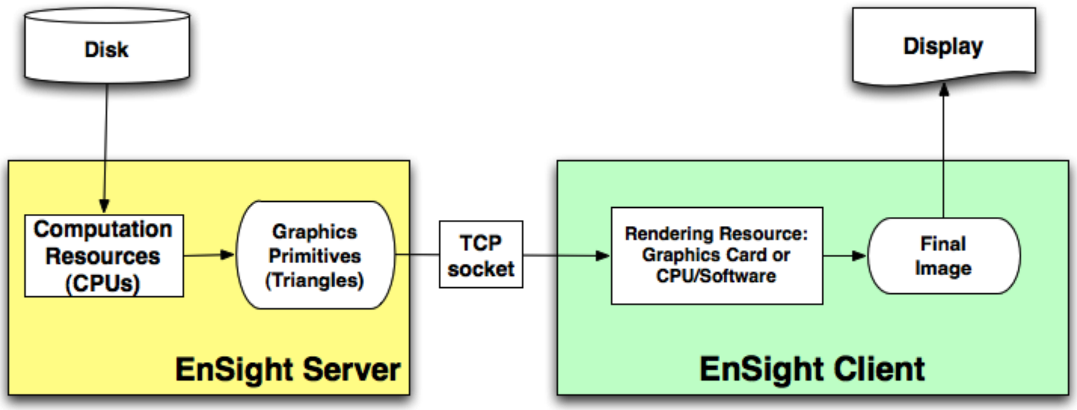
For all but the standalone mode, the ensight client running on your desktop must be able to talk to the ensight server running on the LC clusters. So you must deal with firewall issues, which are different for machine in the Collaboration Zone or the Restricted Zone. Instead of posting elaborate instructions to deal with all of that, we are providing handy wrapper scripts and strongly encourage you to use them. The technically curious can get the details from the scripts themselves. **These instructions will heretofore assume you have downloaded the following tools**. The tools below only work with Mac and Linux machines.

Also it is important to note that **the version number of EnSight on your desktop and the version number on the clusters need to match** if you are using any of the distributed modes. The letter does not matter. So 9.2.2(e) matched 9.2.2(f), but 9.2.2(e) does not match 9.2.1(e). The instructions below assume version 10.0.x. If you are using version 9.2, change "ensight100" to "ensight92."

COLLABORATION ZONE (CZ):
If you will be working on data in the CZ, grab the LLNL EnSight launch utility ensight_desktop_cz.py. This is located at: /usr/global/tools/CEI/LLNL/ceishell/ensight_desktop_cz.tgz. Download and untar this package, and **be sure to keep the three files in it together**. We suggest putting them in $HOME/bin and adding $HOME/bin to your PATH.

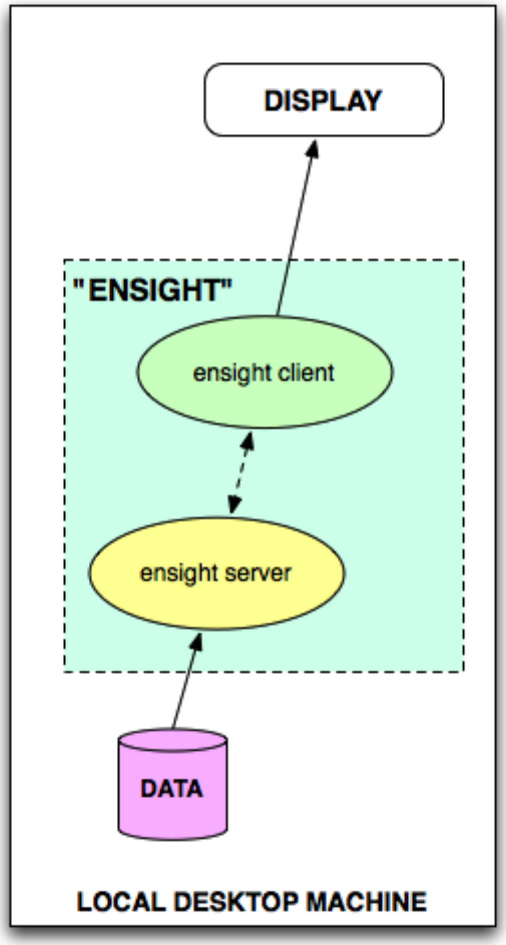
RESTRICTED ZONE (RZ):
If you are working on the RZ, you can simply use /usr/local/bin/ensight_cluster_rz-$version.sh on any RZ cluster, where $version is simply the version of EnSight which will be launched.

EnSight Visualization Pipeline


Disk
Display
Computation Resources (CPUs)
Graphics Primitives (Triangles)
TCP socket
Rendering Resource: Graphics Card or CPU/Software
Final Image
EnSight Server
EnSight Client

To understand EnSight, it's perhaps best to start with the image above, which shows the "visualization pipeline, " which describes how data "moves" from storage on disk to an image on your display: The server reads your data and performs computations on it, such as selection, isosurfacing, mesh creation, etc. The result of these computations is "something that can be drawn," which is generally a collection of graphics primitives, triangles to be specific. These triangles are shipped via a socket (local or network) to the client, which applies zoom, rotation, and other view selection, and then hands the triangles to the graphics card to be rendered into a final image and blitted onto the display.

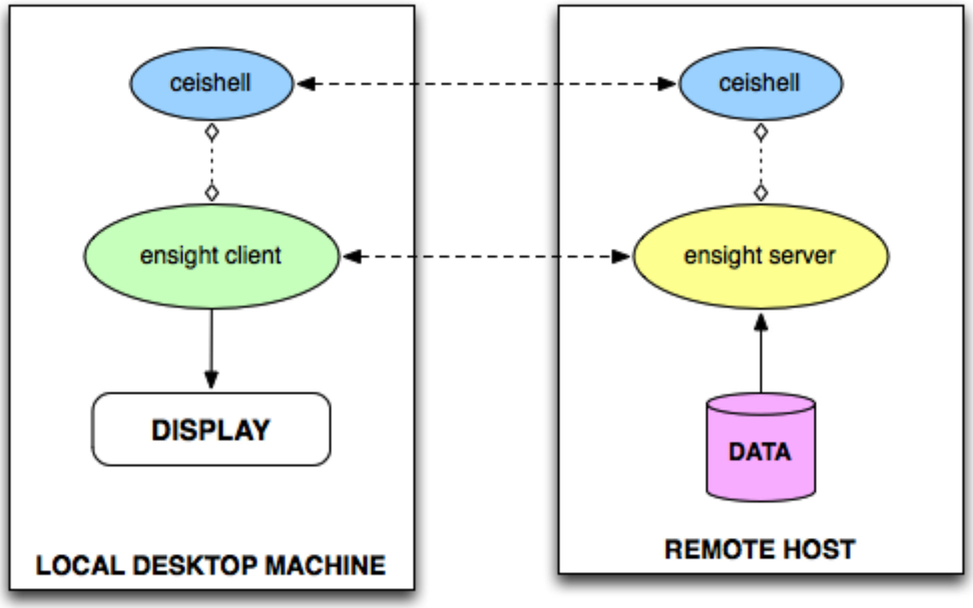
Mode 1: Basic Standalone



In this model, you simply type "ensight100" on the machine where your data lives. In this mode, the client and server are both run on the same machine. **This mode only works if this is your desktop machine and your data is not too huge.** If you have very large data, or are making extremely complex visualizations, you can run out of local memory in this mode.
WARNING: This mode should not be used on the LC machines. Use distributed modes instead. The bottom line is to **only use this mode when your data is on your local desktop workstation .** If standalone mode is used on a login node of an LC machine, EnSight might use lots of CPU resources and this can slow down processing for other users on the login nodes. Probably, at the very least, **your user experience will be greatly diminished in this mode**, as EnSight will have to use GLX, which is a network version of OpenGL, to render its images, which will greatly slow down rendering and interactivity and **can also cause blank screens** and other improper behaviors. As if that were not enough, every element of the GUI will require multiple X11 calls and verifications over the network to draw and respond to, making the GUI unresponsive and more difficult to use. T

To use stand-alone mode, type the following on your desktop:
ensight100

Mode 2: Distributed Serial



In this mode, the data is remote, and the client is local. This is ideal for data generated in large simulations which need to be viewed on your desktop. You launch the server on the remote host, close to the data. It reads the data off disk, creates lots of triangles, which it sends to the client running on your desktop to render quickly and directly using your graphics card for best performance.
Note: Running the client and server both remotely moves the green oval to the remote host but leaves the display on the local machine, causing a "rendering bottleneck" as the client sends countless rendering commands over GLX to your machine. That is a bad thing. Don't do it.

You should run the server on a batch node to avoid using all the CPU and memory on the login node.
The instructions are different for CZ than for RZ.

Note for offsite desktops: You must be able to see the license server to use EnSight. See the Environment section of this web page: https://computing.llnl.gov/vis/ensight.shtml#env

If you are connecting to a cluster on the CZ (hosts that do not have "rz" in their name), do the following. This works for offsite desktop machines too:
Let's assume you are connecting from your linux or mac desktop computer "mydesktop" to LC cluster "cluster" where you have batch job 3322 running with 4 nodes reserved, named

cluster[49-52].
Download ensight_desktop_cz.py (see above)
On your desktop, run ensight_desktop_cz.py cluster --batchnode cluster49 -m
You may be prompted for a password to connect to cluster. After typing your password, look for a message like this:

```
********************************************************
When you see "job started!" then type the following on your desktop:
ensight100 -ceishell "connect://localhost?port=7926&timeout=-1"
********************************************************
```

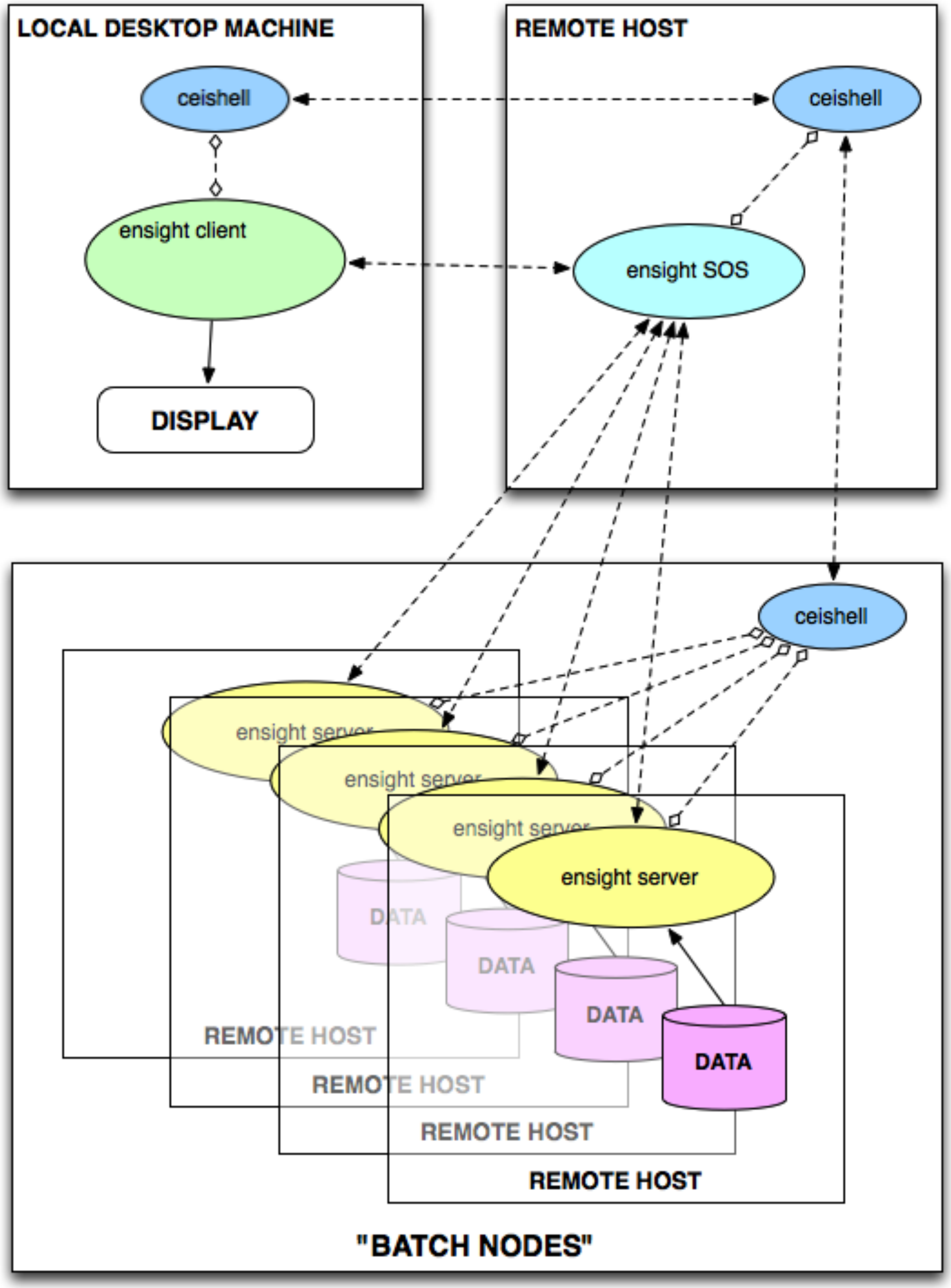
In a separate xterm window, type the given command exactly as given, including quotes, and you should be good to go. It's best to copy and paste it if you can. If you do not see the above message, look carefully for errors. See "troubleshooting" below for common errors. **You can reuse the ceishell connection, in other words, if you exit EnSight, you can start EnSight again and it will use the same underlying connection without needing a new password.**

If you are connecting to a cluster on the RZ (hosts that have "rz" in their name, e.g. rzzeus):
Let's assume you are connecting from your linux or mac desktop computer "mydesktop" to LC cluster "rzcluster" where you have batch job 3322 running with 4 nodes reserved, named rzcluster[49-52].
On the cluster's login node, run /usr/local/bin/ensight_cluster_rz-10.0.2.py mydesktop --batchnode rzcluster49 -m
You may be prompted for a password to connect to your desktop. After typing your password, look for a message like this:

```
********************************************************
When you see "job started!" then type the following on your desktop:
ensight100 -ceishell "connect://localhost?port=7926&timeout=-1"
********************************************************
```

In a separate xterm window on your desktop, type the given command exactly as given, including quotes, and you should be good to go. It's best to copy and paste it if you can. If you do not see the above message, look carefully for errors. See "troubleshooting" below for common errors. **You can reuse the ceishell connection, in other words, if you exit EnSight, you can start EnSight again and it will use the same underlying connection without needing a new password.**

Mode 3: Distributed Parallel SOS



The third mode is a more powerful version of distributed mode, because it's parallel!   In this mode, you run several ensight servers on the batch nodes. Each batch must be able to see your data, obviously. An EnSight Server of Servers (SOS) runs on the login node or one of the batch nodes and acts as an intermediary to direct commands from the client to the servers and send triangles from the servers to the client. The client does all rendering locally on the desktop.

The advantage of this mode is that very large data can in this way be worked on in smaller subsets. This also speeds up the computation and reduces memory overhead for the servers.

The instructions are different for CZ than for RZ.

Note for offsite desktops: You must be able to see the license server to use EnSight. See the Environment section of this web page: https://computing.llnl.gov/vis/ensight.shtml#env

If you are connecting to a cluster on the CZ (hosts that do not have "rz" in their name), do the following. This works for offsite desktop machines too:
Let's assume you are connecting from your linux or mac desktop computer "mydesktop" to LC cluster "cluster" where you have batch job 3322 running with 4 nodes reserved, named cluster[49-52]. Let's say you want 8 servers to run on the batch nodes.
Download ensight_desktop_cz.py (see above)
On your desktop, run ensight_desktop_cz.py cluster --batchnode cluster49 --sos 3322 8 -m
You may be prompted for a password to connect to cluster. After typing your password, look for a message like this:

```
********************************************************
When you see "job started!" then type the following on your desktop:
ensight100 -ceishell "connect://localhost?port=7926&timeout=-1" -sos
********************************************************
```

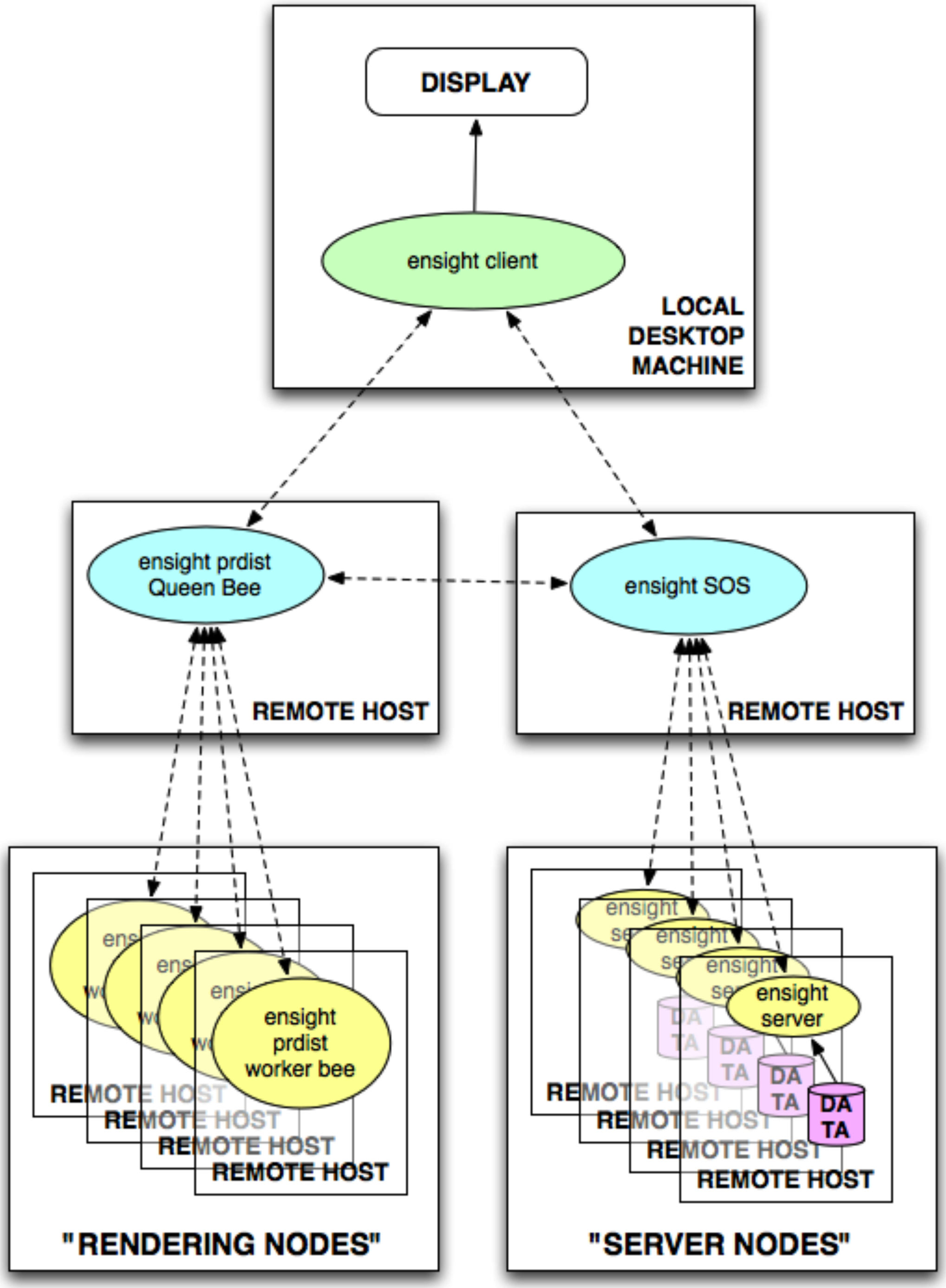
In a separate xterm window, type the given command exactly as given, including quotes, and you should be good to go. It's best to copy and paste it if you can. If you do not see the above message, look carefully for errors. See "troubleshooting" below for common errors. **You can reuse the ceishell connection, in other words, if you exit EnSight, you can start EnSight again and it will use the same underlying connection without needing a new password.**

If you are connecting to a cluster on the RZ (hosts that have "rz" in their name, e.g. rzzeus):
Let's assume you are connecting from your linux or mac desktop computer "mydesktop" to LC cluster "rzcluster" where you have batch job 3322 running with 4 nodes reserved, named rzcluster[49-52]. Let's say you want 8 servers to run on the batch nodes.
On the cluster's login node, run /usr/local/bin/ensight_cluster_rz-10.0.2.py mydesktop --batchnode rzcluster49 --sos 3322 8 -m
You may be prompted for a password to connect to your desktop. After typing your password, look for a message like this:

```
********************************************************
When you see "job started!" then type the following on your desktop:
ensight100 -ceishell "connect://localhost?port=7926&timeout=-1" -sos
********************************************************
```

In a separate xterm window on your desktop, type the given command exactly as given, including quotes, and you should be good to go. It's best to copy and paste it if you can. If you do not see the above message, look carefully for errors. See "troubleshooting" below for common errors. **You can reuse the ceishell connection, in other words, if you exit EnSight, you can start EnSight again and it will use the same underlying connection without needing a new password.**

Mode 4: Distributed Parallel SOS + Distributed Rendering (prdist)



This mode is the same as Mode 3, but handles the case where the servers generate so many triangles that the local client runs out of memory and starts crashing the client on your desktop. This is unusual, but does happen on very large data with billions of zones.

Here we add the "prdist" mechanism, which redirects the triangles from the SOS back to the batch nodes to render. The batch nodes here must have graphics cards, which most of the clusters at LLNL do not have. When the rendering nodes are finished with the triangles, the final result is composed into an image, and only the image is sent to the client. This saves memory on the client and allows the batch nodes each to work on only a subset of the triangles, speeding them up and allowing them to handle large visualizations.

You probably do not need this feature. This mode is cumbersome to set up and manage, and does not yet work with ceishell. We have written a tool called ensightLauncher to handle this mode but it only works with Ensight 6 data. Please consult with lc-graphics@llnl.gov for more information.

Troubleshooting

Here are a few common errors you might see when using the launch script. If you see one not listed, please let us know. lc-graphics@llnl.gov

1. transport_connect:socket_connect: connection refused 107;
Meaning: One component temporarily could not connect to another component.
Solution: This is not necessarily an error! Don't worry about it.

2. srun: error: Unable to allocate resources: Access/permission denied
Meaning: srun failed to launch the servers on the batch node.
Solution: You probably gave the wrong batch job number.

3. error waiting for job
Meaning: something happened earlier that caused things to not work.
Solution: Look earlier in the output for the real cause and deal with that.

4. transport_connect: it appears that port 1109 is in use.
Meaning: You already have a ceishell running on your desktop.
Solution: Use the -m argument to the script.